IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: Hansen, et al. Application No.: 10/757,925 Filed: January 16, 2004 For: METHOD AND SOFTWARE FOR PARTITIONED GROUP ELEMENT SELECTION OPERATION	Examiner: Jesse R. Moll Technology Center/Art Unit: 2181 Confirmation No.: 5116

Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

THIRD DECLARATION OF CRAIG HANSEN

UNDER 37 CFR § 1.131

Sir:

I, Craig Hansen, hereby declare the following to be true:

BACKGROUND

1. I am the same Craig Hansen who submitted a Declaration of Craig Hansen Under 37 CFR § 1.131 dated July 31, 2009 (Hansen I declaration) and a Second Declaration of Craig Hansen Under 37 CFR § 1.131 dated September 17, 2009 (Hansen II declaration).

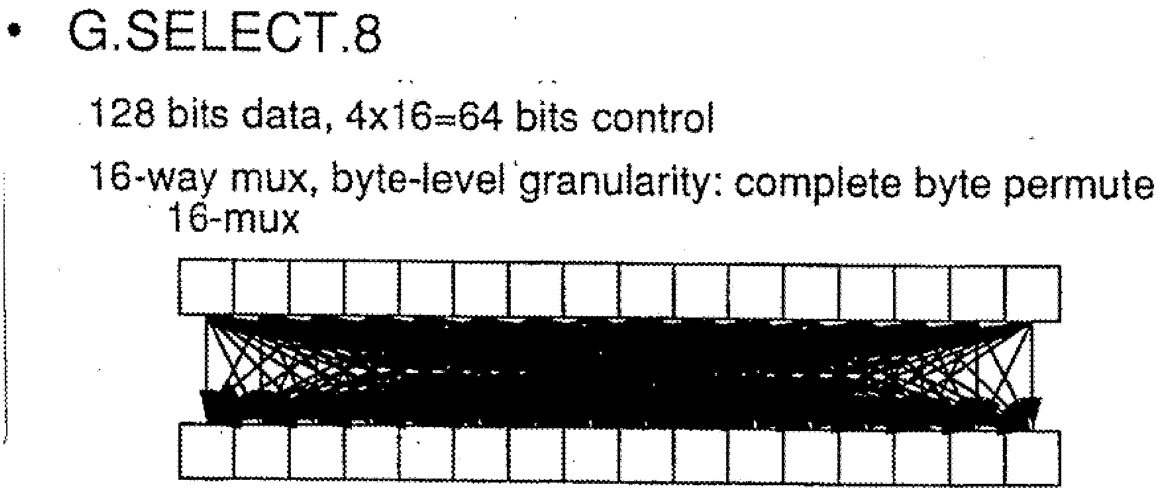
2. I have reviewed the Office Action dated December 23, 2009, in particular paragraph 2 thereof which states:

> Although the G.Select.8 instruction is shown to rearrange data based on a 64-bit selector, there is no evidence showing that the elements are provided in parallel to the catenated result. Additionally regarding claims 12 and 25, while the

documents mention multiply instructions the evidence does not show providing products as a catenated result.

(*original emphasis*)

3. On page MU20396 of Exhibit 1 of my Hansen I declaration, the G.SELECT.8 instruction is disclosed as follows:



- G.SELECT.8

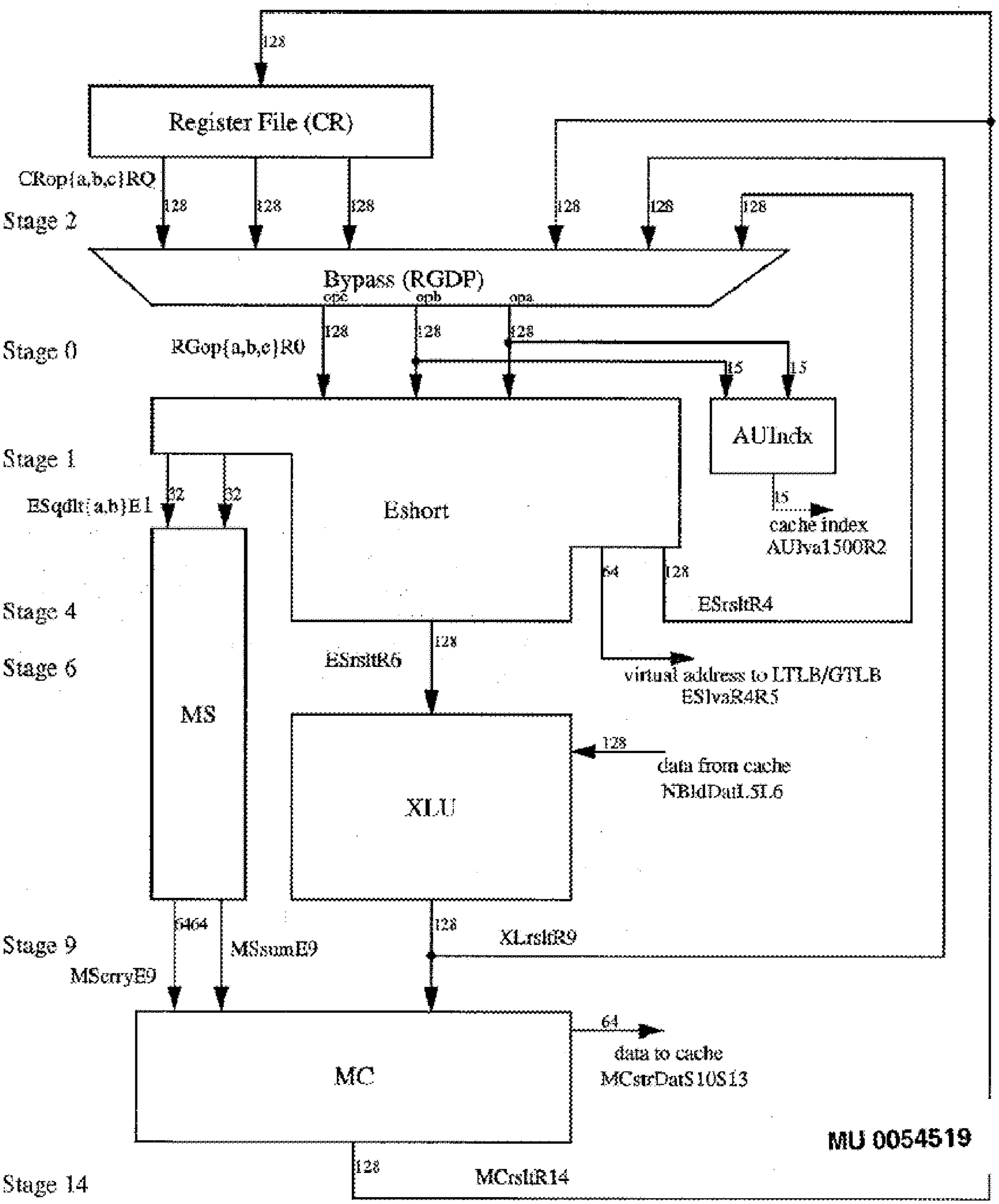
128 bits data, 4x16=64 bits control

16-way mux, byte-level granularity: complete byte permute

4. In this disclosure, the diagram is expressed in the manner of a signal flow diagram, where it would be understood by one skilled in the art that the data signals expressed side-by-side (the boxes) are provided in parallel and that multiplexor circuits shown side-by-side operate in parallel, thus providing the elements in parallel to a catenated result. One skilled in the art would have recognized this disclosure as a full disclosure that the elements of the G.Select.8 operation are provided in parallel to the catenated result, contrary to the conclusion made by the Office Action.

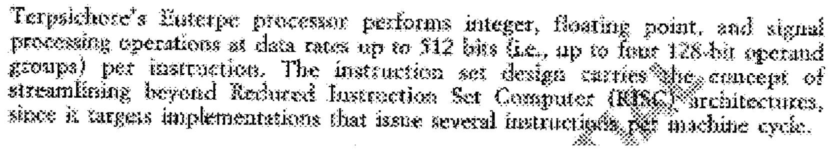
5. On page MU54519 of Exhibit A of my Hansen II declaration, the main pipeline of an implementation of a processor performing a G.Select.8 instruction is disclosed as follows:

Main Pipeline



6. This diagram shows the main pipeline of a processor, where the pipeline stages are indicated according to a vertical position, several of which are marked by the designation “Stage 2,”, “Stage 0”, “Stage 1”, “Stage 4”, “Stage 6”, “Stage 9” and “Stage 14.” It would be understood that each stage represents activity during a single clock cycle, and signals shown side-by-side transit in parallel. Most of the signal flows are oriented downward, representing that an instruction is processed by reading values from the “Register File (CR)”, passing through “Bypass (RGDP)”, “Eshort,” “XLU,” and “MC”, but there are a few upward flows, representing results that can be bypassed (in place of) the register sources of following instructions and also written to the “Register File (CR).” Of particular note is the “XLU” block, where the “ESrsltR6” signal is designated as 128 bits in size and transits to the “XLU” block and the “XLrsltR9” signal is also designated as 128 bits in size and transits from the “XLU” block. It also can be seen that the section of the document from MU54502 to MU54517 “describes the use of the instruction families in the XLU” and in particular MU54514 shows the operation of the G.Select.8 instruction, indicating that the “XLU” block performs the significant data manipulation of the G.Select.8 instruction. Because one can see that the ESrsltR6 and XLrsltR9 signals are 128 bits in size and correspond to the src1[127:0] and dst1[127:0] signals (also 128 bits in size) respectively of MU54514 it would be apparent to one skilled in the art that the catenated result of the operation provides the elements in parallel. One skilled in the art would have recognized this disclosure as a further disclosure that the elements of the G.Select.8 operation are provided in parallel to the catenated result, contrary to the conclusion made by the Office Action.

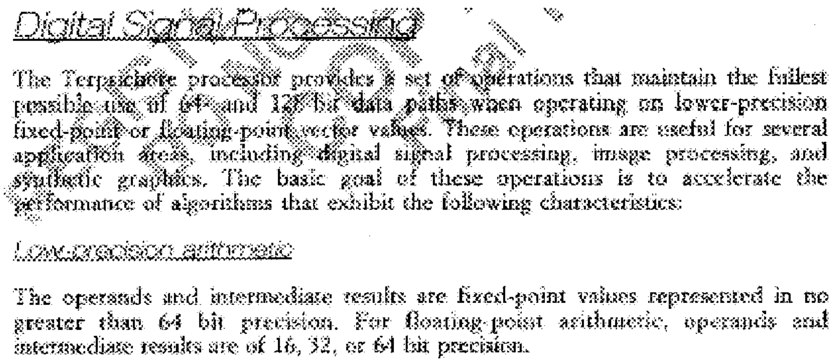
7. On page MU23223 of Exhibit 2 of my Hansen I declaration, the second paragraph is disclosed as follows:



Terpsichore's Euterpe processor performs integer, floating point, and signal processing operations at data rates up to 512 bits (i.e., up to four 128-bit operand groups) per instruction. The instruction set design carries the concept of streamlining beyond Reduced Instruction Set Computer (RISC) architectures, since it targets implementations that issue several instructions per machine cycle.

8. It would be understood by one skilled in the art that RISC architectures have been characterized by the major feature that they are designed to issue an instruction per machine cycle, where previous CISC designs required several machine cycles to perform an instruction. The statement above indicates that the processor of the claimed invention performs signal processing operations at data rates up to 512 bits (i.e., up to four 128-bit operand groups) per instruction. This data rate corresponds to the capabilities of the "Register File" block shown in the previously discussed "Main Pipeline" diagram, showing three 128-bit operands flowing out of the register file and a fourth 128-bit operand flowing into the register file. The disclosed paragraph above states that the design is meant to be streamlined to the point of at least one instruction per machine cycle, where the data rate of catenated results would be at least 128-bits per machine cycle. This adds further detail to the "Main Pipeline" diagram consistently showing that the 128-bit data paths described in the diagram show a 128-bit catenated result provided in parallel. One skilled in the art would have recognized this disclosure as further disclosure that the elements in the G.Select.8 operation are provided in parallel to the catenated result, contrary to the conclusion made by the Office Action.

9. On page MU23228 of Exhibit 2 of my Hansen I declaration, the paragraphs 6 & 7 are disclosed as follows:



Digital Signal Processing

The Terpsichore processor provides a set of operations that maintain the fullest possible use of 64- and 128-bit data paths when operating on lower-precision fixed-point or floating-point vector values. These operations are useful for several application areas, including digital signal processing, image processing, and synthetic graphics. The basic goal of these operations is to accelerate the performance of algorithms that exhibit the following characteristics:

Low-precision arithmetic

The operands and intermediate results are fixed-point values represented in no greater than 64 bit precision. For floating-point arithmetic, operands and intermediate results are of 16, 32, or 64 bit precision.

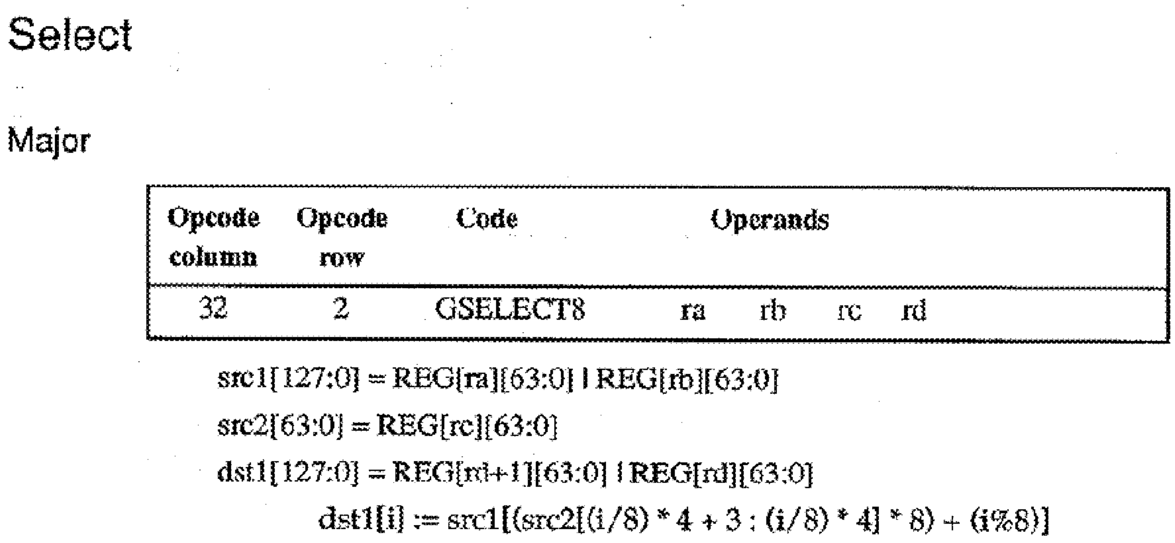
10. Again, this section clearly shows that the purpose of the digital signal processing instructions of the Terpsichore processor is to fully utilize 64-bit and 128-bit data paths of the processor while performing operations on elements of smaller size. The 128-bit data paths to and from the "XLU" block of the Main Pipeline clearly shows the catenated result being provided in parallel in order to fully utilize the 128-bit data path. One skilled in the art would have recognized this disclosure as a further disclosure that the elements of the G.Select.8 operation are provided in parallel to the catenated result, contrary to the conclusion made by the Office Action.

11. The evidence provided above and disclosed in Exhibits 1 and 2 of the Hansen I declaration and Exhibit A of the Hansen II declaration provide sufficient evidence that we conceived of providing in parallel the data elements selected by the fields to respective predetermined positions in a catenated result, as claimed, prior to the August 1, 1995 date for Lee (U.S. Patent No. 6,381,690).

12. In regards to Claim 12 and 25, this evidence also shows providing products of group floating-point multiply operations as a catenated result. On Page MU23326-MU23328 from Exhibit 2 of the Hansen I declaration, group floating-point multiply instructions of various precisions are disclosed. In particular, MU23326-MU23327 lists "GF.MUL.16 group floating-point multiply half," "GF.MUL.32 group floating-point

multiply single," and "GF.MUL.64: group floating-point multiply double" instructions. The instructions are described on MU23327, and the definition section on MU23327 and MU23328 shows an implementation of these instructions where the contents of registers specified by ra and rb are read into variables a and b (a←REG[ra]. B←REG[ra]), and the floating-point multiply operations are performed on fields of the variables a and b, namely ai and bi (ai←F[prec, $a_{i+prec-1+i..i}$], bi←F[prec, $b_{i+prec-1+i..i}$]), producing products in ci (ci ← ai*bi), which are packed into fields of c ($c_{i+prec-1+i..i}$ ← PackF[prec, ci]. Finally, c, containing catenated floating-point products is written into the register specified by rc (REG[rc]←c). The value of c expresses the catenated products in parallel and I have previously shown that the present invention is designed to fully utilize a 128-bit data path, thereby presenting the products to the register file in parallel (MU54519 diagram, particularly MCrsltR14). One skilled in the art would have recognized this disclosure as a full disclosure of "providing the plurality of products to partitioned fields of a result register as a catenated result," as recited in claims 12 and 25, contrary to the conclusion made by the Office Action. This provides sufficient evidence of our conception of claims 12 and 25 prior to the August 1, 1995 date for Lee.

13. As currently presented, claim 1 and other independent claims contain the phrase "the single instruction independently specifying the first register and the second register" requiring that the claimed instruction specify the data operand as the citation of two independently specified registers. On page MU54514 of Exhibit A of my Hansen II declaration, the G.Select.8 instruction is disclosed as follows:



Select

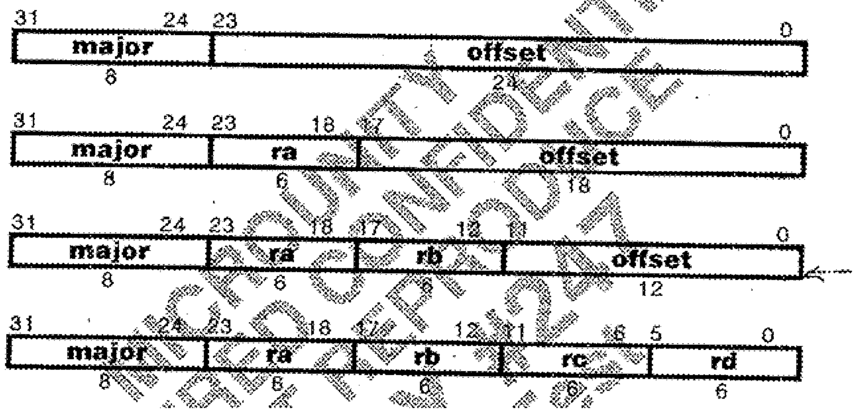
Major

Opcode column	Opcode row	Code	Operands			
32	2	GSELECT8	ra	rb	rc	rd

```
src1[127:0] = REG[ra][63:0] | REG[rb][63:0]
src2[63:0] = REG[rc][63:0]
dst1[127:0] = REG[rd+1][63:0] | REG[rd][63:0]
        dst1[i] := src1[(src2[(i/8) * 4 + 3 : (i/8) * 4] * 8) + (i%8)]
```

The disclosure above shows that the instruction code G.Select.8 specifies four register operands, namely ra, rb, rc, and rd. The src1[127:0] operand (the data operand) is composed of the contents of the register specified by ra (REG[ra][63:0]), catenated (|) to the contents of the register rb (REG[rb][63:0]). On page MU23257 of Exhibit 2 of my Hansen I declaration, instruction formats are disclosed as follows:

The general forms of the instructions coded by a major operation code are one of the following:



The last instruction format shown in the above diagram applies to this G.Select.8 instruction, showing the major opcode in bits 31..24, ra in bits 23..18, rb in bits 17..12, rc in 11..6 and rd in 5..0. The ra and rb operands are separate, independently specified operands that specify the first register and the second register that, catenated together, provide the data operand, thus supporting this aspect of the claimed invention. One skilled in the art would have recognized this disclosure as a full disclosure of the instruction independently specifying the first register and the second register, thereby providing evidence of our conception of the claimed invention prior to the August 1, 1995 date for Lee

14. In my Hansen II declaration, I explained how Exhibit 1 of my Hansen I declaration and Exhibit A of my Hansen II declaration provided evidence of conception prior to the August 1, 1995 date for Lee of "the data selection operand comprising a plurality of fields each independently selecting one of the plurality of data elements," as previously recited in claim 1 and other independent claims. The evidence cited in my Hansen II declaration similarly evidences our conception of "the data selection operand comprising a plurality of fields each selecting any one of the plurality of data elements and each field having a value not restricted by the other fields included in the data selection operand," as recited in claim 1 and other independent claims.

15. I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true, and the these statements are made with knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application, and any patent issuing thereon, or any patent to which this declaration is directed.

22-June-2010
Date

Craig Hansen